UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/591,708	06/09/2000	Stuart J. Jacobs	00-8010	2685

25537 7590 09/05/2007

VERIZON
PATENT MANAGEMENT GROUP
1515 N. COURTHOUSE ROAD
SUITE 500
ARLINGTON, VA 22201-2909

EXAMINER	
HA, LEYNNA A	

ART UNIT	PAPER NUMBER
2135	

NOTIFICATION DATE	DELIVERY MODE
09/05/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patents@verizon.com

Office Action Summary	Application No.	Applicant(s)	
	09/591,708	JACOBS ET AL.	
	Examiner	Art Unit	
	LEYNNA T. HA	2135	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 June 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-6,8-12 and 14-22* is/are pending in the application.

4a) Of the above claim(s) *7,13 and 23* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-6,8-12 and 14-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

ETAILED ACTION

1. Claims 1-6, 8-12, and 8-22 remain pending.

Claims 7, 13, and 23 are cancelled.

Continued Examination Under 37 CFR 1.114

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 6/18/2007 has been entered.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-6, 8-12, and 14-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Minear, et al. (US 5,983,350), and further in view of Mason (US 5,668,998).

As per claim 1:

Minear teaches in a node operative within a network of a plurality of nodes, a method for performing cryptographic-related functions comprising:

executing an application program in a user space at the node; **(col.5, lines 34-53)**

receiving an input requiring cryptographic-related processing; **(col.6, lines 13-27 and col.11, lines 30-33)**

generating a message via the application program based on the input, the message representing one of a predefined set of messages for processing by one of a plurality of cryptographic processing component in a kernel space located within the network node; **(col.6, lines 7-8 and 33-40 and col.7, lines 23-40)**

transmitting the message to one of a socket handler *[and a call handler]* in kernel space at the node to obtain a transmitted message; **(col.7, lines 43-54)**

forwarding the transmitted message to a request handler at the node **(col.12, lines 41-48)** which generates a function call to the cryptographic processing component appropriate for the transmitted message; **(col.11, line 54 – col.12, line 12)**

performing the cryptographic-related processing by the cryptographic processing component appropriate for the transmitted message. **(col.6, lines 9-13)**

Minear discloses the claimed socket handler to obtain a transmitted message in the form of a PF-SADB socket where it is much like a routing socket and is for processing to receive client requests (col.7, lines 27-32 and 43-45). Minear also discloses the claimed request handler that generates a function call to the cryptographic

processing component as the required information is stored in a request structure that is bound to the IP datagram being processed where the request is of type crypto_request_t (col.11, lines 64-66). Minear further discusses one function, the cyl_enqueue_request is used to initiate either an encrypt or a decrypt action where this function is designed to be called by the cryptographic engine interface (col.12, lines 41-43). Minear discloses communications between workstations to an unprotected network or to another system (col.5, lines 34-58) that involves transmitted messages being received (col.6, lines 35-37). This obviously suggests calls being handled on the sending and receiving sides during communications across the networks. However, Minear did not explain a call handler to obtain a transmitted message.

Mason discloses an invention that presents an API which maps the DICOM standard services menu onto a framework of service interface objects which perform a selected DICOM services within or between PACS networks (col.1, line 66 – col.2, line 3). Mason explains that each service interface object is uniquely associated with a user handler and a provider handler wherein the association with a user/provider handler pair enable the pair to "handle" communications for the associated service interface object (col.2, lines 7-9). Mason further discusses the user handler enables an application to send a user-specified DICOM-encoded file message to a specified destination using the association control protocol. A provider handler receives the user-specified DICOM-encoded SEND message from an open association, stores the data set in a file, and reports to the requesting application (call back), upon completion of the requested

service (col.13, lines 15-22). Hence, user/provider handler obviously suggests a call handler to obtain a transmitted message.

Therefore, it would have been obvious for a person of ordinary skills in the art to combine the teaching of Minear with Mason to teach a call handler as a user/provider handler pair because this enables the pair to handle communications for the associated message of the service request (Mason – col.2, lines 7-9 and col.13, lines 15-22).

As per claim 2: **See Minear on col.4, lines 1-28 and col.11, lines 46-53;** discusses the method of claim 1, wherein the cryptographic-related processing includes at least one of: verifying or generating a digital signature; encrypting data; decrypting data, retrieving a digital certificate or certificate revocation list; retrieving, verifying the hierarchy, and self-signed certificate processing within the node; or certificate age checking.

As per claim 3: **See Minear on col.6, line 66 – col.7, line 6 and 26-45;** discusses the method of claim 1, wherein transmitting includes: generating a user datagram (UDP) message containing an identifier associated with a predetermined cryptographic related functions and transmitting the UDP message via a UDP socket to the socket handler.

As per claim 4: **See Minear on col.11, line 54 – col.12, line 22;** discusses the method of claim 1 generating an output message via the application program wherein the output message requiring cryptographic-related processing, transmitting based on the required cryptographic-related processing, one of the predefined set of messages to the cryptographic processing component; performing the cryptographic-related processing, and outputting the processed message.

As per claim 5:

Minear teaches a computer readable medium having stored thereon a plurality of sequences of instructions that may be invoked by a plurality of predefined messages, said instructions including sequences of instructions which, when executed by a processor in a user space, cause said processor to perform a method comprising:

receiving an input representing one of predefined messages; **(col.6, lines 13-27 and col.11, lines 30-33)**

transmitting, based on the input, a function call representing a request for cryptographic related processing to a cryptographic processing module executed by the process; and **(col.11, line 54 – col.12, line 12)**

performing the cryptographic-related processing by the cryptographic processing in a kernel space; **(col.6, lines 8-27 and col.11, lines 30-33)**

wherein at least the receiving the transmitting and the performing are implemented by public key authentication infrastructure (PKAI) comprising: **(col.3, line 57 – col.4, line 28)**

user space components including a user application program, a PKAI control daemon, a certificate database, a PKAI operations daemon and a PKAI remote server daemon; and **(col.5, lines 60-63 and col.6, lines 7-12 and 32-48)**

kernel space components including PKAI socket handler **(col.7, lines 43-54)**, *[a PKAI call handler]* and a PKAI request handler; **and col.11, line 54 – col.12, line 12 and lines 41-48) (col., lines)**

wherein certain of the user space components communicate with other of the user space components and certain of the kernel space components communicate with other of the kernel space components; and **(col.5, lines 34-55 and col.6, lines 7-12 and 32--40)**

wherein other certain of the user space components communication with other certain of the kernel space components. **(col.7, lines 23-40)**

Minear discloses the claimed socket handler to obtain a transmitted message in the form of a PF-SADB socket where it is much like a routing socket and is for processing to receive client requests (col.7, lines 27-32 and 43-45). Minear also discloses the claimed request handler that generates a function call to the cryptographic processing component as the required information is stored in a request structure that is bound to the IP datagram being processed where the request is of type crypto_request_t (col.11, lines 64-66). Minear further discusses one function, the cyl_enqueue_request is used to initiate either an encrypt or a decrypt action where this function is designed to be called by the cryptographic engine interface (col.12, lines 41-43). Minear discloses communications between workstations to an unprotected network or to another system (col.5, lines 34-58) that involves transmitted messages being received (col.6, lines 35-37). This obviously suggests calls being handled on the sending and receiving sides during communications across the networks. However, Minear did not explain a call handler to obtain a transmitted message.

Mason discloses an invention that presents an API which maps the DICOM standard services menu onto a framework of service interface objects which perform a

selected DICOM services within or between PACS networks (col.1, line 66 – col.2, line 3). Mason explains that each service interface object is uniquely associated with a user handler and a provider handler wherein the association with a user/provider handler pair enable the pair to "handle" communications for the associated service interface object (col.2, lines 7-9). Mason further discusses the user handler enables an application to send a user-specified DICOM-encoded file message to a specified destination using the association control protocol. A provider handler receives the user-specified DICOM-encoded SEND message from an open association, stores the data set in a file, and reports to the requesting application (call back), upon completion of the requested service (col.13, lines 15-22). Hence, user/provider handler obviously suggests a call handler to obtain a transmitted message.

Therefore, it would have been obvious for a person of ordinary skills in the art to combine the teaching of Minear with Mason to teach a call handler as a user/provider handler pair because this enables the pair to handle communications for the associated message of the service request (Mason – col.2, lines 7-9 and col.13, lines 15-22).

As per claim 6: See Minear on col.4, lines 1-28 and col.11, lines 46-53; discusses the method of claim 5, wherein the cryptographic-related processing includes at least one of: verifying or generating a digital signature; encrypting data; decrypting data, retrieving a digital certificate or certificate revocation list; retrieving, verifying the hierarchy, and self-signed certificate processing within the node; or certificate age checking.

As per claim 7: Cancelled

As per claim 8: **See Minear on col.3, line 58 – col.4, line 28;** discussing the input represents a digitally signed network control message requiring verification.

As per claim 9:

Minear discloses a cryptographic module, comprising:

a memory configured to store a plurality of cryptographic processing programs in user space on a computer-readable medium, each program being invoked via one of a plurality of predefined messages; and **(col.5, lines 34-53 and col.6, lines 7-12)**

a processor configured to:

receive an input requiring cryptographic-related processing, **(col.6, lines 7-8 and 33-40 and col.7, lines 23-40)**

generates one of predefined messages based on the input, **(col.6, lines 13-27 and col.11, lines 30-33)**

transmit the message to the first one of the cryptographic processing programs, and to perform, in kernel space, the cryptographic-related processing; **(col.7, lines 43-54 and col.11, line 54 – col.12, line 12)**

wherein the module receives, generates, transmits and performs through infrastructure comprising:

user space components including a user application program, a control daemon, a certificate database, a operations daemon and a remote server daemon; and **(col.5, lines 60-63 and col.6, lines 7-12 and 32-48)**

kernel space components including socket handler **(col.7, lines 43-54)**, *[a call handler]* and a request handler; **and col.11, line 54 – col.12, line 12 and lines 41-48) (col., lines)**

wherein certain of the user space components communicate with other of the user space components and certain of the kernel space components communicate with other of the kernel space components; and **(col.5, lines 34-55 and col.6, lines 7-12 and 32-40)**

wherein other certain of the user space components communication with other certain of the kernel space components. **(col.7, lines 23-40)**

Minear discloses the claimed socket handler to obtain a transmitted message in the form of a PF-SADB socket where it is much like a routing socket and is for processing to receive client requests (col.7, lines 27-32 and 43-45). Minear also discloses the claimed request handler that generates a function call to the cryptographic processing component as the required information is stored in a request structure that is bound to the IP datagram being processed where the request is of type crypto_request_t (col.11, lines 64-66). Minear further discusses one function, the cyl_enqueue_request is used to initiate either an encrypt or a decrypt action where this function is designed to be called by the cryptographic engine interface (col.12, lines 41-43). Minear discloses communications between workstations to an unprotected network or to another system (col.5, lines 34-58) that involves transmitted messages being received (col.6, lines 35-37). This obviously suggests calls being handled on the

sending and receiving sides during communications across the networks. However, Minear did not explain a call handler to obtain a transmitted message.

Mason discloses an invention that presents an API which maps the DICOM standard services menu onto a framework of service interface objects which perform a selected DICOM services within or between PACS networks (col.1, line 66 – col.2, line 3). Mason explains that each service interface object is uniquely associated with a user handler and a provider handler wherein the association with a user/provider handler pair enable the pair to "handle" communications for the associated service interface object (col.2, lines 7-9). Mason further discusses the user handler enables an application to send a user-specified DICOM-encoded file message to a specified destination using the association control protocol. A provider handler receives the user-specified DICOM-encoded SEND message from an open association, stores the data set in a file, and reports to the requesting application (call back), upon completion of the requested service (col.13, lines 15-22).. Hence, user/provider handler obviously suggests a call handler to obtain a transmitted message.

Therefore, it would have been obvious for a person of ordinary skills in the art to combine the teaching of Minear with Mason to teach a call handler as a user/provider handler pair because this enables the pair to handle communications for the associated message of the service request (Mason – col.2, lines 7-9 and col.13, lines 15-22).

As per claim 10: See Minear on col.4, lines 1-28 and col.11, lines 46-53; discusses the method of claim 9, wherein the cryptographic-related processing includes at least one of: verifying or generating a digital signature; encrypting data; decrypting

data, retrieving a digital certificate or certificate revocation list; verifying a certificate's hierarchy; self-signed certificate processing; retrieving, verifying and storing a digital certificate; or certificate age checking.

As per claim 11: See Minear on col.5, lines 34-55 and Mason on col.2, lines 7-9; discusses the method of claim 9, wherein when transmitting the message, the processor is further configured to: transmit a function call to the first cryptographic processing program.

As per claim 12: See Minear on col.12, lines 1-16; discusses the method of claim 9, wherein the processor is further configured to: transmit the result of the cryptographic-related processing to an application program.

As per claim 13: Cancelled

As per claim 14:

Minear discusses a method of performing cryptographic-related functions in a node coupled to other nodes in a network, the node includes an application program executed in user space for handling communications with the other nodes the method comprising:

receiving in said node an input requiring cryptographic-related processing; **(col.6, lines 13-27 and col.11, lines 30-33)**

generating in said node a predefined message based on the input, the message one of a plurality of predefined message usable by of the cryptographic processing programs executed by one of a plurality of cryptographic processing component in a kernel space, each one of said messages being associated with a respective one of

said cryptographic-related functions; **(col.6, lines 7-8 and 33-40 and col.7, lines 23-40)**

transmitting in said node a predefined message to a socket handler in kernel space or a call handler in kernel space to obtain a transmitted message; **(col.7, lines 43-54)**

forwarding the transmitted message to a request handler within the node **(col.12, lines 41-48)** which generates a function call to the cryptographic processing component appropriate for the transmitted message; **(col.11, line 54 – col.12, line 12)**

performing in said node, via cryptographic processing program the required cryptographic-related operation. **(col.6, lines 9-13)**

Minear discloses the claimed socket handler to obtain a transmitted message in the form of a PF-SADB socket where it is much like a routing socket and is for processing to receive client requests (col.7, lines 27-32 and 43-45). Minear also discloses the claimed request handler that generates a function call to the cryptographic processing component as the required information is stored in a request structure that is bound to the IP datagram being processed where the request is of type crypto_request_t (col.11, lines 64-66). Minear further discusses one function, the cyl_enqueue_request is used to initiate either encrypt or decrypt action where this function is designed to be called by the cryptographic engine interface (col.12, lines 41-43). Minear discloses communications between workstations to an unprotected network or to another system (col.5, lines 34-58) that involves transmitted messages being received (col.6, lines 35-37). This obviously suggests calls being handled on the

sending and receiving sides during communications across the networks. However, Minear did not explain a call handler to obtain a transmitted message.

Mason discloses an invention that presents an API which maps the DICOM standard services menu onto a framework of service interface objects which perform a selected DICOM services within or between PACS networks (col.1, line 66 – col.2, line 3). Mason explains that each service interface object is uniquely associated with a user handler and a provider handler wherein the association with a user/provider handler pair enable the pair to "handle" communications for the associated service interface object (col.2, lines 7-9). Mason further discusses the user handler enables an application to send a user-specified DICOM-encoded file message to a specified destination using the association control protocol. A provider handler receives the user-specified DICOM-encoded SEND message from an open association, stores the data set in a file, and reports to the requesting application (call back), upon completion of the requested service (col.13, lines 15-22). Hence, user/provider handler obviously suggests a call handler to obtain a transmitted message.

Therefore, it would have been obvious for a person of ordinary skills in the art to combine the teaching of Minear with Mason to teach a call handler as a user/provider handler pair because this enables the pair to handle communications for the associated message of the service request (Mason – col.2, lines 7-9 and col.13, lines 15-22).

As per claim 15: See Minear on col.12, lines 1-16; discusses the method of claim 14, returning the result of the performing to the application program.

As per claim 16: See Minear on col.4, lines 1-28 and col.11, lines 46-53; discusses the method of claim 14, a request for digital signature generation, a request for digital signature verification, a request for data encryption, a request for data decryption, a request for retrieval of digital certificate, a request verification of a certificate's hierarchy, a request for self-signed certificate processing, a request for certificate age checking.

As per claim 17: See Minear on col.11, lines 46-53 and col.12, lines 17-19; discusses the method of claim 16, wherein the request for digital signature generation includes a request for at least one of the RSA signature, secret key MD5 signature generation, elliptic curve signature generation or digital signature standard signature generation.

As per claim 18: See Minear on col.3, line 58 – col.4, line 28; discusses the method of claim 16, wherein the request for digital signature verification includes a request for at least one of the RSA signature verification, secret key MD5 signature verification, elliptic curve signature verification or digital signature standard signature verification.

As per claim 19: See Minear on col.11, lines 46-53 and col.12, lines 17-21; discusses the method of claim 16, wherein the request for data encryption includes a request for at least one of the RSA encryption or elliptic curve encryption.

As per claim 20: See Minear on col.4, lines 1-28; discusses the method of claim 16, wherein the request for data decryption includes a request for at least one of the RSA decryption or elliptic curve decryption.

As per claim 21: **See Minear on col.10, lines 35-60;** discusses the method of claim 14, wherein the performing includes accessing a remote server via the network to retrieve cryptographic related information.

As per claim 22:

Minear discloses a computer-readable medium that stores instructions executable in user space by at least one processor in kernel space to perform a method for providing cryptographic-related functions, the method comprising:

receiving in at least one processor a first function call from a predefined list a first function call from a predefined list of function calls representing available cryptographic-related functions executable by the at least one processor; **(col.6, lines 13-27 and col.11, lines 30-33)**

generating in at least one processor in the environment a request message based on the first function call, a for cryptographic processing to further transmit the request message representing a request for processing by a cryptographic processing module executed by the at least one processor; **(col.7, lines 43-54 and col.11, line 54 – col.12, line 12)**

transmitting in at least one processor the request message to the cryptographic processing module; and **(col.12, lines 41-47)**

performing in at least one processor the cryptographic-related function;

wherein the receiving, generating, transmitting and performing through infrastructure comprising:

user space components including a user application program, a control daemon, a certificate database, a operations daemon and a remote server daemon; and **(col.5, lines 60-63 and col.6, lines 7-12 and 32-48)**

kernel space components including socket handler **(col.7, lines 43-54)**, *[a call handler]* and a request handler; **and col.11, line 54 – col.12, line 12 and lines 41-48) (col., lines)**

wherein certain of the user space components communicate with other of the user space components and certain of the kernel space components communicate with other of the kernel space components; and **(col.5, lines 34-55 and col.6, lines 7-12 and 32-40)**

wherein other certain of the user space components communication with other certain of the kernel space components. **(col.7, lines 23-40)**

Minear discloses the claimed socket handler to obtain a transmitted message in the form of a PF-SADB socket where it is much like a routing socket and is for processing to receive client requests (col.7, lines 27-32 and 43-45). Minear also discloses the claimed request handler that generates a function call to the cryptographic processing component as the required information is stored in a request structure that is bound to the IP datagram being processed where the request is of type crypto_request_t (col.11, lines 64-66). Minear further discusses one function, the cyl_enqueue_request is used to initiate either encrypt or decrypt action where this function is designed to be called by the cryptographic engine interface (col.12, lines 41-43). Minear discloses communications between workstations to an unprotected network

or to another system (col.5, lines 34-58) that involves transmitted messages being received (col.6, lines 35-37). This obviously suggests calls being handled on the sending and receiving sides during communications across the networks. However, Minear did not explain a call handler to obtain a transmitted message.

Mason discloses an invention that presents an API which maps the DICOM standard services menu onto a framework of service interface objects which perform a selected DICOM services within or between PACS networks (col.1, line 66 – col.2, line 3). Mason explains that each service interface object is uniquely associated with a user handler and a provider handler wherein the association with a user/provider handler pair enable the pair to "handle" communications for the associated service interface object (col.2, lines 7-9). Mason further discusses the user handler enables an application to send a user-specified DICOM-encoded file message to a specified destination using the association control protocol. A provider handler receives the user-specified DICOM-encoded SEND message from an open association, stores the data set in a file, and reports to the requesting application (call back), upon completion of the requested service (col.13, lines 15-22). Hence, user/provider handler obviously suggests a call handler to obtain a transmitted message.

Therefore, it would have been obvious for a person of ordinary skills in the art to combine the teaching of Minear with Mason to teach a call handler as a user/provider handler pair because this enables the pair to handle communications for the associated message of the service request (Mason – col.2, lines 7-9 and col.13, lines 15-22).

As per claim 23: Cancelled

Response to Arguments

4. Applicant's arguments with respect to claims 1-6, 8-12, and 8-22 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to LEYNNA T. HA whose telephone number is (571) 272-3851. The examiner can normally be reached on Monday - Thursday (7:00 - 5:00PM).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Vu can be reached on (571) 272-3859. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

LHa

KIM VU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100